DYNAMIC BANDWIDTH ALLOCATION AND CONGESTION CONTROL SCHEMES FOR VOICE AND DATA MULTIPLEXING IN WIDEBAND PACKET TECHNOLOGY

K. Sriram
AT&T Bell Laboratories

Room 3H-607
Crawfords Corner Road
Holmdel, New Jersey 07733

ABSTRACT

Wideband Packet Technology integrates packetized voice and data messages using statistical multiplexing. The statistical multiplexer should make efficient use of the transmission bandwidth while meeting the performance (e.g., delay and packet loss) requirements for all traffic types. We describe here a novel bandwidth allocation method called the (T_1, T_2) - scheme which efficiently integrates packetized voice and data traffic. Voice and data packets are queued separately, and the (T_1, T_2) - scheme is used to facilitate dynamic bandwidth sharing and mutual overload protection between the two queues. T_1 and T_2 are time limits for transmitting voice and data packets continually while the voice and data queues are visited alternately. The scheme guarantees bandwidths to voice and data in the proportion of their respective time slice allocations, T_1 and T_2. However, the bandwidth allocations are flexible in the sense that whenever one queue is exhausted, the transmission is immediately moved over to the other queue if it has a packet waiting to be served. In addition, the system also implements a voice block dropping scheme in which the less significant bits in voice packets are dropped during periods of congestion. We present results based on a simulation model which illustrate that the two schemes together provide the desired performance in terms of (1) very good voice quality, (2) low delay and packet loss, (3) efficient use of transmission bandwidth, and (4) protection in overload.

1. INTRODUCTION

In an integrated voice and data network, the resources such as switching nodes and wideband transmission links are efficiently shared by voice and data for improved cost effectiveness. In this paper we propose a novel scheme called the (T_1, T_2) - scheme which is flexible in allocating bandwidth to voice and data, and also lends itself to the implementation of a graceful overload control scheme based on embedded ADPCM in which the less significant bits of voice are discarded in periods of congestion. We first present a literature review, and motivate the need for developing the (T_1, T_2) - scheme for voice and data multiplexing.

In the literature, several schemes have been proposed for integrated multiplexing of voice and data on wideband transmission links. Two commonly known schemes are (1) the movable boundary scheme (see [1] [2] [3] [4] [5] and references therein), and (2) the burst switching scheme[6] [7] [8]. Some useful variations and refinements of the first scheme have also been proposed and analyzed by various authors [9] [10] [11] [12] [13] [14]. In the movable boundary scheme, voice and data traffic dynamically share the channel capacity on a wideband transmission link. Multiplexing is done within synchronous time-division multiplexed (TDM) frames. Frame duration is fixed and equals the voice packetization interval. Each frame is divided into a number of time slots of equal duration. Voice and data packets are of equal duration and either can be accommodated in a time slot. A predetermined number of time slots, N_d, in each frame is reserved for data transmission so that surges of voice traffic do not cause excessive delays for data traffic. The rest of the time slots, N_v, in a frame are shared by voice and data traffic with voice given priority over data. Voice packets are not queued except over a frame interval. However, data packets are queued until they are served.

In the burst switching scheme, a burst may be either a voice burst or a data burst. A voice burst is a whole voice talkspurt. Each burst has a header providing its destination and other information. Thus the multiplexing and switching functions occur at the burst level. The frames on a transmission link are of the same duration, which is divided into a fixed number of time slots. Each time slot accommodates one byte. A synchronous time slot, i.e., one time slot per frame, is regarded as a channel. Voice is encoded at 64 kbit/s rate using pulse code modulation (PCM). When a voice burst arrives, the multiplexer allocates a channel to that voice burst if a channel is available. If there is contention, a voice burst is buffered up to a few milliseconds. If a channel does not become available within that time, then the burst experiences front-end clipping. The clipping continues until a channel becomes available. A data burst, on the other hand, cannot be clipped. Hence, it is stored in a queue until a channel becomes available.

The two currently known schemes described above for voice and data multiplexing have serious disadvantages in that they impose certain restrictions on the way packetization and bandwidth allocations are done. Further, they do not facilitate the implementation of variable rate coding or bit-dropping techniques on voice packets[15] [16] [17] [18] [19] [20] [21] [22], and do not use bandwidth as efficiently as it can be done. A detailed discussion of these methods and their relative merits and demerits is given in [23].

Bandwidth allocation is a critical issue in a statistical multiplexer for packetized voice and data. The statistical multiplexer should make efficient use of the transmission bandwidth while meeting the performance (e.g., delay and packet loss) requirements for all traffic types. Ideally, each traffic type should be protected from overloads caused by the other traffic in the multiplexer. Further, the multiplexer should have some congestion control schemes to allow for graceful degradation of performance when overloads occur. We describe here a novel (T_1, T_2) - scheme for multiplexing voice and data which meets the above mentioned objectives. This scheme overcomes the disadvantages of the previously described schemes, and also allows voice, voiceband data, and facsimile to be coded by a range of bit rates using PCM, standard ADPCM, embedded ADPCM, etc. The (T_1, T_2) - scheme works synergistically with a congestion control scheme based on embedded coding[20] and bit-dropping (or block-dropping) on voice packets[18][19].

The dynamic bandwidth allocation and congestion control scheme are described in detail in Section 2. In Section 3, we present results based on a simulation model to illustrate the performance of the schemes. For some analytical results related to some limiting cases of the (T_1, T_2) - scheme see Leung and Eisenberg[24]. The work of Coffman et al.[25] was also motivated by the (T_1, T_2) - scheme proposed in this paper.

The concepts presented here were developed as part of the Wideband Packet Technology project at the AT&T Bell Laboratories, and have been implemented in the Integrated Access and Cross-connect System (IACS) [26] [27] [28] [29] [30] [31]. A component of the IACS is the Integrated Access Terminal (IAT) which performs the integration and packet multiplexing of signaling, voice, and data traffic. An overview of the design and performance of the IAT is presented in Sriram et al[32]. Detailed analysis and results related to some aspects of the IAT performance are reported in [18][20] [33] [34] [35] [36] [37].

CH2829-0/90/0000-1003 $1.00 © 1990 IEEE

2. *DESCRIPTION OF THE DYNAMIC BANDWIDTH ALLOCATION AND CONGESTION CONTROL SCHEMES*

2.1 *THE (T_1,T_2) - SCHEME FOR DYNAMIC BANDWIDTH ALLOCATION*

In Fig. 1, the multiplexer includes three packet queues used for receiving and storing three different types of traffic: signaling, voice and data. Signaling packets, voice traffic packets and data traffic packets, stored in the buffers, remain therein until they are selected out by a (T_1,T_2) - scheme. Subsequently the packets are transmitted out of the block dropping congestion controller onto a transmission link.

Referring to the (T_1, T_2) - packet selector in Fig.1, the signaling packets are given highest priority, and are selected shortly after they are received in the signaling queue. This guarantees that the signaling packets experience almost no delay and zero packet loss. When there are no packets in the signaling queue, either voice or data packets are served according to the (T_1, T_2) - scheme. First the waiting voice packets are served until a maximum of T_1 ms or until the voice queue is exhausted, whichever occurs first. Then the data packets are served likewise with a corresponding time slice allocation of T_2 ms. Both voice and data queues are subject to interruptions (upon completion of the packet in service) to serve any signaling packets that may have been received in the signaling queue while the service for either voice or data was in progress. During such excursions to the signaling queue, the T_1/T_2 timer for the voice/data queue is suspended. And the timer is resumed when the signaling queue is exhausted and service is returned to the interrupted voice or data queue. Fig. 2 illustrates some sample representations of the operation of the (T_1,T_2) - scheme.

The voice and data transmission intervals T_1 and T_2 are each of the order of a few milliseconds, i.e., comparable to a few multiples of typical voice or data packet transmission times. With existing VLSI technology the switch-over time from one queue to another is very small (a few tens of microseconds) as compared to a typical packet transmission time (a few hundred microseconds). If there are no more packets to be served in a queue currently in service, then service is immediately switched over to the other queue. Thus each traffic type is allowed to use any spare bandwidth that may be momentarily available due to inactivity of the other.

Usually the signaling traffic intensity is very small compared to aggregate voice and data traffic intensities. Therefore, the (T_1, T_2) - scheme essentially allocates the bandwidth on the transmission link of Fig. 1 to the aggregate voice and data traffic in the ratio of T_1 to T_2. In other words, this scheme guarantees a minimum bandwidth of $\{T_1 /(T_1 +T_2)\}C$ for the aggregate voice traffic and $\{T_2/(T_1 +T_2)\}C$ for the aggregate data traffic, where C is the overall transmission capacity of the link (signaling traffic is assumed to use only a negligible portion of C). Thus the priority scheme provides protection to each type of traffic so long as that traffic remains within its guaranteed bandwidth. The values of the transmission intervals T_1 and T_2 for selecting from the voice and data queues, respectively, can be selected to accommodate packet delay requirements for voice and data traffic. A duration of the voice transmission interval T_1 that is much larger than the data transmission interval T_2 will decrease delays for voice packets at the expense of increased delays for data packets, and vice versa. The values of the intervals T_1 and T_2 can be chosen either to reserve certain minimum bandwidth proportions for voice and data or to adjust delays for voice and data packets, as required.

2.2 *THE VOICE BLOCK DROPPING SCHEME FOR CONGESTION CONTROL*

Before proceeding to describe the block dropping congestion controller shown in Fig. 1, we will first describe how voice is coded and organized into packets, and how packets are organized into different blocks containing bits of different order of significance. Additional details regarding the voice packetization protocol and coding schemes can be found in [19][20].

Fig. 3 shows how a voice packet is organized. Each voice source is sampled at an 8 kHz rate and is encoded using an embedded ADPCM scheme at a 32 kbit/s rate. The four-bit voice samples from an interval are collected and organized into a packet. As shown in Fig. 3, the sample bits are reorganized into four blocks according to bit significance, and a header is attached to the front of the packet. All of the least significant bits from the samples are put into block 1, the next more significant bits are put into block 2, and the two most significant bits are put into blocks 3 and 4, respectively. It may be noted here that the speech coding rate in the IAT is optional: 64 kbit/s PCM and 40 kbit/s embedded ADPCM are also among the options available. The packetization interval is fixed, and hence the packet size would vary depending on the coding scheme used. The voice packet header incorporates a range of information about the packet, such as its destination, time-stamp, and other protocol related information. The header also contains information regarding the initial and the current number of droppable blocks in the packet. This information is used for packet delineation and block dropping decision at intermediate nodes, and for decoding and speech reconstruction at the final destination. A speech classifier is used to distinguish a voiceband data burst from a voice burst.

Sriram and Lucantoni [18] presented a description and analysis of block dropping and its traffic smoothing effects in a packet voice multiplexer. The following is a description of the block dropping congestion control algorithm used in the integrated voice and data multiplexer of Fig. 1. Here we use the expression "block dropping" instead of "bit dropping" which was used in our related previous publications[18][35][37].

Fig. 4 shows a flow chart representing the operation of the block dropping congestion controller of Fig. 1. If the packet is a voice packet, then the block dropping algorithm is invoked. A congestion measure F is obtained by taking a weighted sum of the number of packets waiting in the voice and data queues, VQ and DQ. Thus the congestion measure F is computed as follows:

$$F = \alpha VQ + \beta \min(DQ, DQ^*). \qquad (1)$$

DQ^* in (1) is a limiting value on DQ just in the computation of this congestion measure F. The reason for this is to protect voice from excessive block dropping in situations when there is possibility of data being excessively bursty and DQ too large as a result. Also, at a given user location, it may be known a priori that the volume of voice traffic is very low relative to the volume of data traffic. In such a situation, block dropping from the voice packets is not an effective solution for the congestion. Therefore, one may simply set the data queue cap DQ^* to zero so that voice block dropping remains unaffected by the heavy data traffic. The parameters α and β are the weights on the voice and data queues, VQ and DQ, respectively. Thus α and β measure the relative influence allowed for voice and data queues in the block dropping congestion control scheme. The values of α and β are normally selected to be "one" each. In such a case, the congestion measure F is simply the sum of the two queues, i.e., $F = VQ + \min(DQ, DQ^*)$. However, as we will illustrate in the numerical examples, the values of α and β may be tuned to suit a particular traffic scenario or to meet certain performance objectives. For example, when voice traffic volume is very low relative to data, then β could be set to zero to protect voice block dropping from data congestion.

If the congestion measure F is smaller than the lower block dropping threshold $B1$, the voice packet is transmitted intact (see Fig. 4). If the congestion measure F exceeds $B1$ but is less than the upper block dropping threshold $B2$, then the first droppable block in the voice packet (block 1) is dropped and the remainder of the voice packet is transmitted. The first droppable block contains the least significant bits of the voice samples. If the congestion measure F exceeds the upper block dropping threshold $B2$ also, then the first and the second blocks (block 1 and block 2) are both dropped.

On the transmission link, a packet with blocks dropped requires less transmission time than a full packet would require. This enables the multiplexer to deplete the voice packets from the voice queue VQ at a faster rate during the critical periods of congestion. This in turn helps reduce the delays for voice and data packets waiting in the queues of the memories. Detailed descriptions of performance evaluation of the objective and subjective effects of the block dropping on voice quality can be found in Sriram and Lucantoni[18], Karanam et al.[35], Bowker and Dvorak[36], and Dravida and Sriram[37].

3. *SIMULATION MODEL, NUMERICAL EXAMPLES, AND DISCUSSION*

3.1 DESCRIPTION OF THE SIMULATION MODEL

The discrete-event simulation model for the multiplexer is written in FORTRAN and run on a Cray-1. We let the simulation run for about 15 minutes of real-time operation of the multiplexer. Typically, one to two million voice and data packets are serviced in the multiplexer during a 15 minute period. The performance measures of interest are sampled at intervals of a minute, starting at the 10 minute epoch. The resulting six samples are averaged. We repeated the 15 minute runs several times with different seed values in each run for all the random number generators, and found that the results were very close to each other. The simulation program used in this study evolved from the the study reported in [33], and the confidence intervals are very narrow (see Figs. 6,7 in [33]). The confidence intervals are very narrow for two reasons: (i) we simulate a very large number (one to two million) of packet arrivals in each simulation run, and (ii) the voice block dropping mechanism significantly smooths the burstiness in the combined voice and data packet arrival process (by effectively increasing the voice packet service rate during periods of congestion in the queues).

Each voice source is simulated by generating a geometrically distributed number of packets at equal intervals during a talkspurt, followed by an exponential silence duration (see Sriram and Whitt[33] and Sriram and Lucantoni[18] for details). The data traffic is simulated by independent exponential or hyper-exponential packet inter-arrival times, and fixed or geometrically distributed packet size. The parameters of the hyper-exponential distribution are varied to simulate data traffic with different burstiness characteristics. The burstiness measure used here is the squared co-efficient of variation, c^2, which is defined as follows:

$$c^2 = \frac{var(X)}{\{E(X)\}^2} \qquad (2)$$

where X is random variable representing the packet inter-arrival time. Let $(p,\ 1-p)$ denote the probabilities of the two states of a hyper-exponential distribution, and let (ξ_1, ξ_2) be the corresponding data packet arrival rates in the two states. Let $\theta = \frac{\xi_1}{\xi_2}$ be the ratio of the two arrival rates. Then the overall arrival rate, λ, and the squared co-efficient of variation, c^2, of this arrival process are given by

$$\lambda = \{p\, \xi_1^{-1} + (1-p)\, \xi_2^{-1}\}^{-1} \qquad (3)$$

$$c^2 = \frac{2\{p + (1-p)\,\theta^2\}}{\{p + (1-p)\,\theta\}^2} - 1 \qquad (4)$$

The values $p = 0.95$, $\theta = 40$ give $c^2 = 17.6$, and $p = 0.9$, $\theta = 10$ give $c^2 = 5.04$. Once p and θ are fixed, the overall arrival rate and packet size are selected according to the desired percentage offered load value.

Traffic and System Parameter Values:

Speech activity occupies on the average about 28% to 42% of the total connection time; the variation in the activity factor is user population dependent, and is due to culture and language differences across the globe. Two different cases of speech activity rates are considered in the numerical examples: 22 packets per second (pps) and 26.25 pps, corresponding to 35% and 42% activity factors, respectively. The mean speech talkspurt and silence lengths are assumed to be 352 ms and 650 ms for the first case, and 420 ms and 580 ms for the second case, respectively. We use a speech activity factor at 35% in Figs. 5-12, and 42% in Figs. 13-15. These speech parameter values are based on some measurements done at the AT&T Bell Laboratories[38] [39], and are the same as those used in our previous studies[18][35]. The size of a regular voice packet is assumed to be 74 bytes including a 10 byte header. In practice, the header size may be smaller but we chose 10 bytes to incorporate some conservatism into our performance predictions. The data packet size is assumed to be geometrically distributed with mean 60 bytes (by default); in some cases it is assumed to be deterministic as indicated. The block dropping thresholds $B1$ and $B2$ are assumed to be 20 and 40, respectively. The α and β parameters in the block dropping algorithm are each set to a value one, unless specified otherwise. The cap on the data queue DQ^* is assumed to be infinite.

The effect of finite buffers is also captured in the simulations. Voice packet loss is estimated by observing the frequency of discarding a voice packet (just prior to its transmission) because its waiting time in the queue exceeded some value, say 20 ms. For the practical range of loads, the voice packet loss probability is seen to be negligible. This is because block dropping on voice packets prevents voice packet loss up to very high loads (also see related discussion in [18]). For the data traffic, we estimate the packet loss probability by observing the frequency of discarding a data packet because its delay exceeded a prespecified value. The maximum queueing delay for a data packet at a node should typically be in the ball park of a few tens of milliseconds while the end-to-end network delay objective may be about 100 to 200 ms. In practice, the buffer sizes for voice and data queues should be in the neighborhood of 60 and 120 packets, respectively. These buffer sizes correspond to approximately 20 ms and 40 ms in terms of time required for emptying a full buffer at the DS1 speed, based on an average packet size of approximately 60 bytes for voice and data. The link transmission rate is assumed to be 1.536 Mbit/s, i.e., the DS1 rate.

3.2 NUMERICAL EXAMPLES AND DISCUSSION

FIFO vs. (T_1, T_2) - Scheme:

In Figs. 5 and 6, the mean bits per sample and the mean delay for voice are compared for the FIFO and the (T_1,T_2) - service disciplines. In the FIFO discipline, voice and data are entered into one queue and serviced on a first-in first-out basis. >From Fig. 5 we note that when data traffic causes overload, the voice traffic is well protected under the (T_1,T_2) - scheme whereas the voice quality rapidly degrades under the FIFO scheme. Fig. 5 also illustrates that voice performance degradation is much worse under the FIFO discipline when the data traffic is bursty (with $c^2 = 17.6$). >From Figs. 5 and 6 we observe that the (T_1,T_2) - scheme provides protection to voice traffic in terms of the mean bits per sample performance at the expense increased data delays (when data causes overload).

It appears that neither a first-in first-out (FIFO) service discipline nor an exhaustive non-preemptive priority scheme is appropriate for multiplexing voice and data packets on a transmission link. In the FIFO case, neither traffic type is protected from overloads caused by the other traffic, and therefore the network would have to be traffic engineered conservatively to meet the performance requirements of the more stringent traffic type. On the other hand if voice traffic is given exhaustive priority over data so as to keep its delay small, the data traffic may suffer unduly long delays and packet losses.

Choice of T_1, T_2 Values:

Figs. 7 through 10 further illustrate how the choice of the (T_1,T_2) parameters influence the performance of the system for a scenario where the data load is fixed while the voice traffic goes into overload. Let f_1 and f_2 denote the fractions of the total link bandwidth, C, allocated to voice and data traffic, respectively. Hence, we have

$$f_i = \frac{T_i}{(T_1 + T_2)}\ , \quad C_i = f_i\, C\ ; \quad i=1,2. \qquad (5)$$

where C_1 and C_2 are approximately the minimum guaranteed bandwidths allocated to voice and data, respectively. Let ρ_1 and ρ_2 denote the voice and data traffic utilizations, respectively, as fractions of the total link transmission capacity (e.g., 1.536 Mb/s for a DS1 link). In Figs. 7 and 8 we see that by allowing f_2 to be larger than the data traffic utilization (ρ_2), the mean delay and the packet loss probabilities for data can be significantly lowered at the expense of only a slight penalty to the mean bits per sample and the mean delay for voice (see Figs. 9,10). When data utilization ρ_2 is larger than f_2, data packets often have to wait through several cycles of T_1 followed by T_2, and this causes data packet delay and loss probability to be high (see Figs. 7,8). On the other hand, when f_2 is larger than ρ_2, data packets tend to get cleared out of the buffers during the T_2 part of each T_1-T_2 cycle. This reduces the data delays (see Fig. 7) to values that are less than the cycle length, $T_1 + T_2$. The dramatic improvement of data performance (for the case $f_2 > \rho_2$ in comparison to the case of $f_2 < \rho_2$) occurs at the expense of only a mild degradation in voice performance (Figs. 9,10) but this is to be expected. The reason is that, in this case, while increasing the bandwidth allocated to data improves the data performance, it takes away very little from the average bandwidth available to voice. In addition, the traffic smoothing due to voice block dropping also helps in sustaining voice performance.

Advantage to Data due to Voice Block Dropping:

Figs. 11 and 12 illustrate the effects of varying the block dropping weighting parameters α and β. It is seen in Fig. 11 that the packet loss performance for data can be significantly improved by setting β to value one so that data can influence block dropping based on its own backlog. Similarly, the corresponding packet delay performance (for data) also improves for $\beta = 1$ in comparison to $\beta = 0$. The resulting reduction in mean bits per sample is again seen to be fairly mild (Fig. 12). This effect once again can be explained in terms of the smoothing effects of block dropping.

Effects of Burstiness of Data:

Note that the speech activity factor for the examples in Figs. 13-15 is set at 42% (see Section 3.1). Figs. 13-15 illustrate the variations in performance as a function of burstiness of the data traffic. These figures show that by a proper choice of T_1,T_2 values, the delays for voice can be maintained small in spite of the burstiness of data. Also the smoothing effect of block dropping is evident in terms of allowing the voice and data delays to increase linearly rather than exponentially when the system goes into overloads. The decrease in the value of the tail probability for data in severe overload (see Fig. 15) is due to reduction of variance of data delay which in turn is due to the load reduction and traffic smoothing caused by block dropping.

4. *CONCLUSIONS*

We described a (T_1,T_2) - scheme which efficiently integrates packetized voice and data traffic. It facilitates dynamic bandwidth sharing and mutual overload protection between voice and data queues. The scheme guarantees bandwidths to voice and data in the proportion of their respective time slice allocations, T_1 and T_2. However, the bandwidth allocations are flexible in the sense that whenever one queue is exhausted, the transmission is immediately moved over to the other queue if it has a packet waiting to be served. In addition, the system also implements a block dropping scheme wherein, during periods of congestion, the less significant bits of voice packets are dropped. We presented results based on a simulation model to illustrate the performance of the schemes. The numerical examples illustrate that the (T_1,T_2) - scheme provides considerable flexibility in bandwidth allocation. The scheme thereby allows us to meet the disparate performance requirements for voice and data, and to increase the efficiency of transmission bandwidth usage. The block dropping congestion control scheme reduces packet losses as well as delays for both voice and data. The subjective effects on voice quality due to block dropping have not been discussed here, but are reported in [20][35][36].

More elaborate description and discussions of this work are available in an extended version of this paper[23]. Extensions of the concepts presented here to broadband ATM networks will be reported in forthcoming papers[40] [41].

5. *ACKNOWLEDGEMENTS*

The author wishes to acknowledge A. Anastasio for design and development of a FORTRAN-based software tool for the simulation experiments done in this study.

REFERENCES

1. K. Kummerle, "Multiplexer performance for integrated line and packet switched traffic," *Int. Conf. Comput. Commun. Record,*, Stockholm, Sweden, August 1974, pp. 507-515.

2. P. Zafiropolo, "Flexible multiplexing for networks supporting line-switched and packet-switched data traffic," in *Int. Conf. Comput. Commun. Record,*, Stockholm, Sweden, August 1974, pp. 517-523.

3. G. J. Coviello and P. A. Vena, "Integration of circuit/packet switching by a SENET concept," *Proc. of Nat. Telecom. Conf.*, vol.2, New Orleans, LA, December 1975, pp. 42.12-42.17.

4. M. Schwartz, *Telecommunication Networks: Protocols, Modeling and Analysis*, Addison-Wesley, 1987, pp.686-715.

5. K. Sriram, P. K. Varshney, and J. G. Shanthikumar, "Discrete-time analysis of integrated voice-data multiplexers with and without speech activity detectors," *IEEE Journal on Selected Areas in Commun.*, vol. SAC-1, no. 6, pp. 1124-1132, Dec. 1983.

6. S. R. Amstutz, "Burst switching—A method for distributed and integrated voice and data switching," *IEEE Commun. Mag.*, pp. 36-42, Nov. 1983.

7. P. O'Reilly, "Performance Analysis of Data in Burst Switching," *IEEE Trans. Commun.*, vol. COM-34, December 1986, pp. 1259-1263.

8. S. R. Amstutz, "Burst Switching - An Update," *IEEE Commun. Magazine*, September 1989, vol. 27, pp. 50-57.

9. I. Gitman, H. Frank, B. Occhiogrosso, and W. Hsiech, "Issues in integrated network design," *Proc. of ICC*, Chicago, June 1977, pp. 38.1.36-38.1.43.

10. J. W. Forgie and A. G. Nemeth, "An efficient packetized voice/data network using statistical flow control," *Proc. of ICC*, Chicago, June 1977, vol. 3, pp. 38.2.44-38.2.48.

11. H. Miyahara and T. Hasegawa, "Integrated switching with variable frame and packet," *Proc. of ICC*, Toronto, Canada, June 1978, vol. 2, pp. 20.3.1-20.3.5.

12. B. Maglaris and M. Schwartz, "Performance evaluation of a variable frame multiplexer for integrated switching networks," *IEEE Trans. Commun.*," vol. COM-29, pp.800-807, June 1981.

13. J. G. Shanthikumar, P. K. Varshney, and K. Sriram, "Priority Cutoff Flow Control Scheme for Integrated Voice/Data Multiplexers," *ACM-Sigmetrics Performance Evaluation Review*, vol. 11, no. 3, pp. 8-14, Fall 1982.

14. N. Janakiraman, B. Pagurek, and J. E. Neilson, "Performance analysis of an integrated switch with fixed or variable frame rate and movable voice/data boundary," *IEEE Trans. Commun.*, January 1984, pp. 34-39.

15. T. Bially, B. Gold and S. Seneff, " A technique for adaptive voice flow control in integrated packet networks," *IEEE Trans. Commun.*, Vol. COM-28, March 1980, pp. 325-333.

16. D. J. Goodman, "Embedded DPCM for variable bit rate transmission," *IEEE Tran. on Commun.*, COM-28, No. 7, July 1980.

17. Y. Yatsuzuka, "High-Gain Digital Speech Interpolation with ADPCM Encoding," *IEEE Tran. on Commun.*, COM-30, No. 7, April 1982, pp. 750-761.

18. K. Sriram and D. M. Lucantoni, "Traffic Smoothing Effects of Bit Dropping in A Packet Voice Multiplexer," Proc. of the *IEEE INFOCOM'88*, New Orleans, March 1988, pp.759-770. Also in the *IEEE Trans. on Commun.*, July 1989, pp. 703-712.

19. M. H. Sherif, R. J. Clark, and G.P. Forcina, "CCITT/ANSI Voice Packetization Protocol," *International J. on Satellite Commun.*, (to appear).

20. M. H. Sherif, G. Bertocci, D. O. Bowker, B. A. Orford, and G. A. Mariano "Overview of G.EMB," *Proceedings of the IEEE Supercomm/ICC'90*, Atlanta, GA, April 15-19, 1990.

21. N. Yin, S. Q. Li, and T. E. Stern, "Congestion Control for Packet Voice by Selective Packet Discarding," *Proc. of GLOBECOM'87*, Tokyo, Japan, November 1987, vol. 3, pp.1782-1786.

22. D. W. Petr, L. A. DaSilva, Jr., and V. S. Frost, "Priority discarding of speech in integrated packet networks," *IEEE J. Selected Areas in Commun.*, June 1989, pp. 644-656.

23. K. Sriram, "Dynamic Bandwidth Allocation and Congestion Control Schemes for an Integrated Voice and Data Network," submitted to the *IEEE Trans. on Commun.*.

24. K. K. Leung and M. Eisenberg, "A Single-Server Queue with Vacations and Gated Time-Limited Service," *Proc. of IEEE INFOCOM'89*, Ottawa, Canada, April 1989, pp. 897-906.

25. E. G. Coffman, G. Fayolle, and I. Mitrani, "Two Queues with Alternating Service Periods," in *Performance '87*, North-Holland (Publ.), 1988, pp. 227-240.

26. R. W. Muise, T. J. Schonfeld and G. H. Zimmerman, "Experiments in wideband packet technology," *Proc. Zurich Seminar on Digital Communication*, March 1986, pp. D4.1-D4.5.

27. D. Sparrell, "Wideband Packet Technology," *Proc. of IEEE GLOBECOM'88*, Hollywood, Florida, November 1988, pp.1612-6.

28. A. H. Daecher, "On the road to Universal Information Services (UIS): Wideband Packet Technology," *Proc. of ENTELEC'89*, pp. 56-59, New Orleans, LA, March 19-22, 1989.

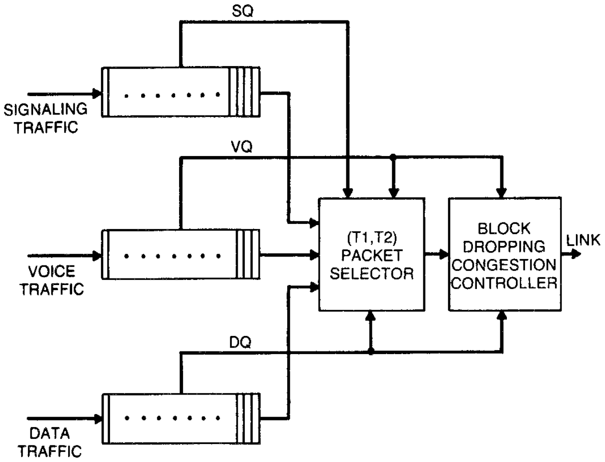

FIGURE 1. Block diagram of an integrated voice and data multiplexer.

29. M. H. Sherif, F. D. Fite and M. C. Gruensfelder, "On the road to Universal Information Services (UIS): The Integrated Access Terminal," *Proc. of ENTELEC'89*, pp. 42-38, New Orleans, LA, March 19-22, 1989.

30. S. B. Andrews, E. J. Messerli, and G. W. R. Luderer, "Faster Packet for Tomorrow's Telecommunications", *AT&T Technology - Products, Systems and Services*, Volume 3, Number 4, 1988, pp. 24-33.

31. G. W. R. Luderer, J. J. Mansell, E. J. Messerli, R. E. Staehler, A. K. Vaidya, "Wideband Packet Technology for Switching Systems", *Proc. of International Switching Symposium*, Phoenix, Arizona, March 1987.

32. K. Sriram, M. A. Gonzalo, D. O. Bowker, and A. U. Mac Rae, "An Integrated Access Terminal for Wideband Packet Networking: Design and Performance Overview," submitted to *ISS'90*, Stockholm, Sweden, May 1990.

33. K. Sriram and W. Whitt, "Characterizing superposition arrival processes in packet multiplexers for voice and data," *IEEE Journal on Selected Areas in Commun.*, vol. SAC-4, no. 6, September 1986, pp. 833-846.

34. H. Heffes and D. M. Lucantoni, "A Markov-modulated characterization of packetized voice and data traffic and related statistical multiplexer performance," *IEEE Journal on Selected Areas in Commun.*, vol. SAC-4, no. 6, September 1986, pp. 856-868.

35. V. R. Karanam, K. Sriram, and D. O. Bowker, "Performance Evaluation of Variable Bit Rate Voice in Packet Networks," *AT&T Technical Journal*: Special Issue on Performance Modeling and Analysis, September-October 1988, pp. 41-56. Also in part in the *Proc. of GLOBECOM'88*, Hollywood, Florida, November 1988, pp.1617-1622.

36. D. O. Bowker and C. A. Dvorak, "Speech Transmission Quality of Wideband Packet Technology," *Proc. of IEEE GLOBECOM'87*, Tokyo, Japan, November 1987, pp.1887-1889.

37. S. Dravida and K. Sriram, "End-to-End Performance Models for Variable Bit Rate Voice over Tandem Links in Packet Networks," *Proc. of INFOCOM'89*, Ottawa, Canada, April 1989, pp. 1089-1097. Also in the *IEEE Journal on Selected Areas in Communications*: Special Issue on Packet Speech and Video, June 1989, pp. 718-728.

38. C. E. May and T. J. Zebo, unpublished paper, AT&T Bell Laboratories, December 1981.

39. D. O. Bowker, unpublished work, AT&T Bell Laboratories, Holmdel, New Jersey, 1987.

40. K. Sriram, "Methodologies for packet multiplexing, bandwidth allocation, and congestion avoidance in broadband ATM networks," to be published.

41. K. Sriram and R.S. McKinney, "Voice packetization and compression in broadband ATM networks," submitted to the *IEEE JSAC*.

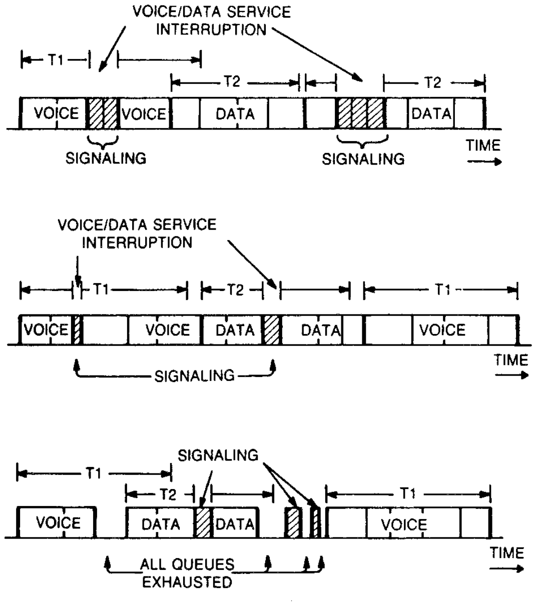

FIGURE 2. Exemplary time lines for operation of the (T1, T2) - scheme.

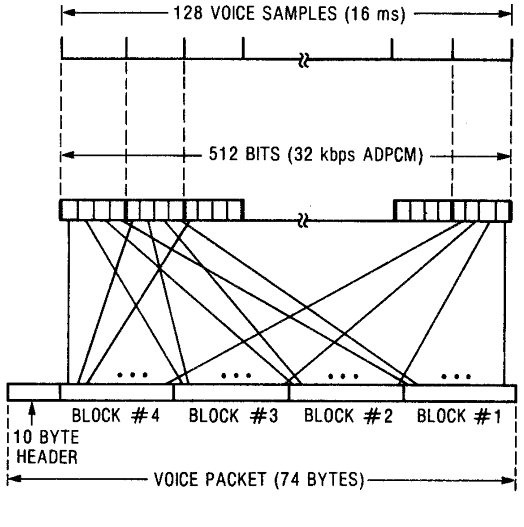

FIGURE 3. Organization of a voice packet.

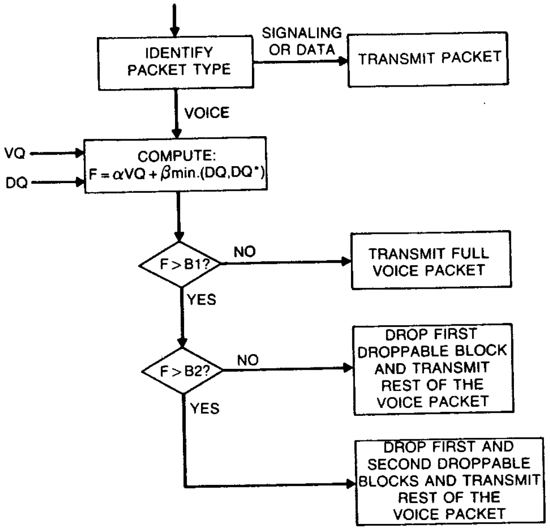
IDENTIFY PACKET TYPE
SIGNALING OR DATA
TRANSMIT PACKET
VOICE
VQ
DQ
COMPUTE: F = αVQ + βmin.(DQ,DQ*)
F>B1?
NO
YES
TRANSMIT FULL VOICE PACKET
F>B2?
NO
YES
DROP FIRST DROPPABLE BLOCK AND TRANSMIT REST OF THE VOICE PACKET
DROP FIRST AND SECOND DROPPABLE BLOCKS AND TRANSMIT REST OF THE VOICE PACKET

FIGURE 4. Flow chart describing operation of the block dropping scheme.

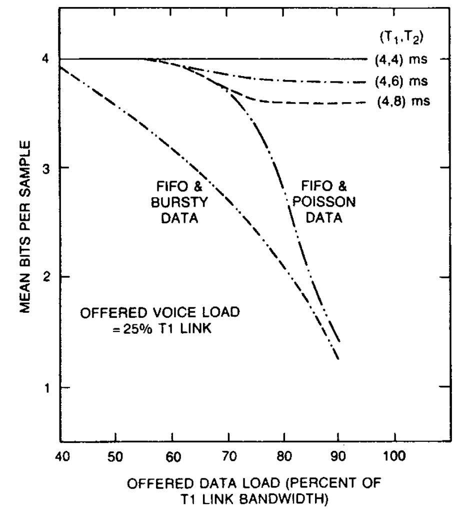

FIGURE 5. Comparison between FIFO and (T1, T2) - scheme for voice service quality in overload.

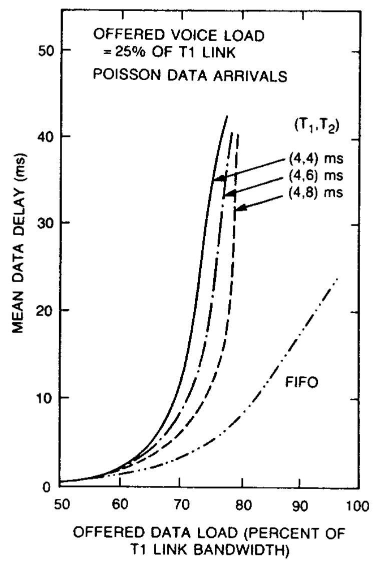

FIGURE 6. Comparison between FIFO and (T1, T2) - scheme for data delay.

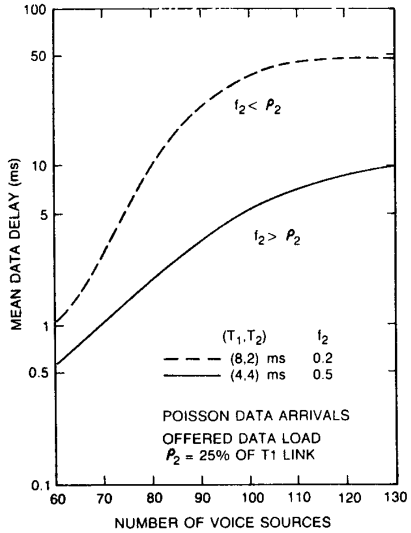

FIGURE 7. Effect of varying bandwidth allocation fractions on mean delay for data.

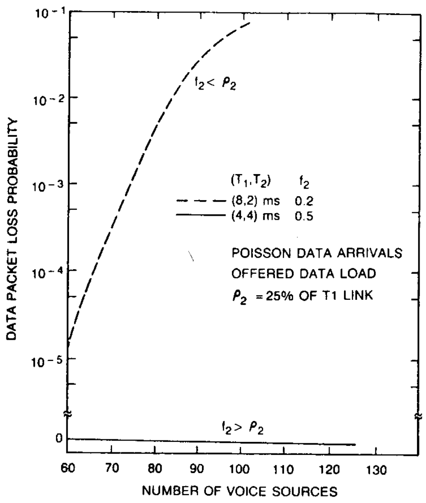

FIGURE 8. Effect of varying bandwidth allocation fractions on probability of data packet loss. (Data packet is lost/dropped when its delay exceeds 60 ms.)

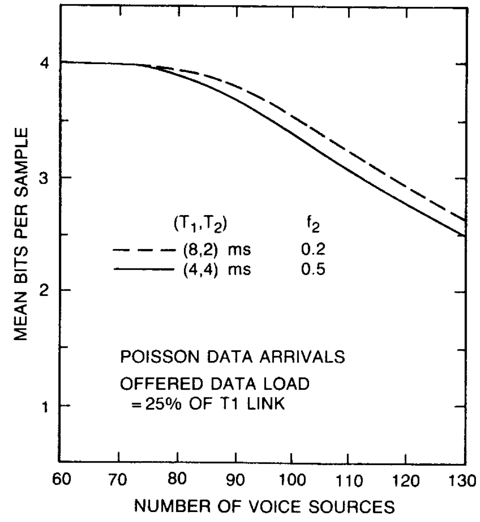

FIGURE 9. Effect of varying bandwidth allocation fractions on voice mean bits per sample.

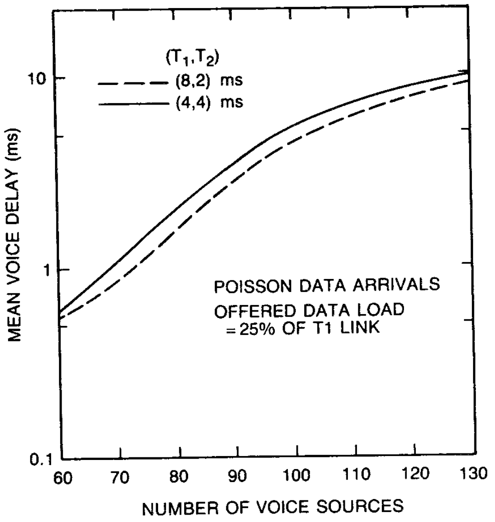

FIGURE 10. Effect of varying bandwidth allocation fractions on mean delay for voice.

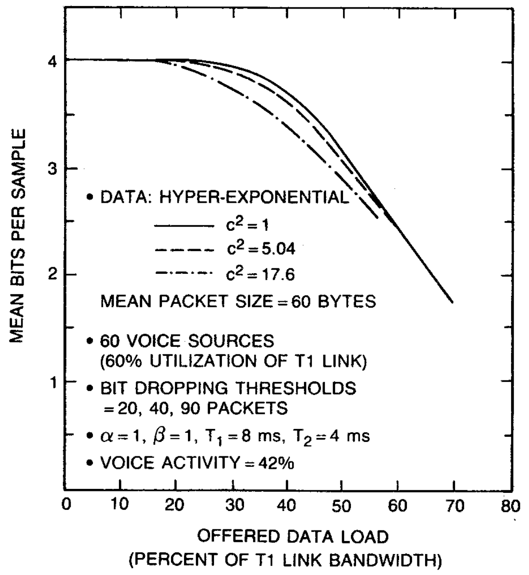

FIGURE 13. Effect of data burstiness on mean bits per sample for voice.

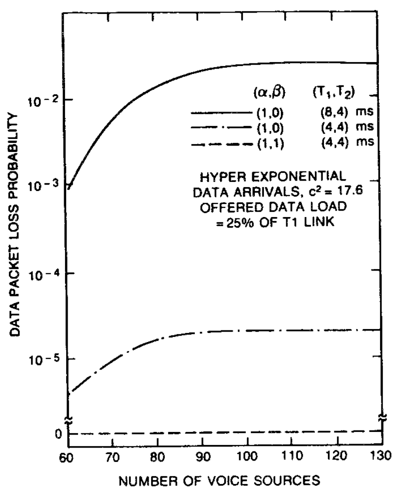

FIGURE 11. Sensitivity of data packet loss probability to block-dropping parameter, i.e., β = 0 vs. β = 1. (Data packet is lost/dropped when its delay exceeds 60 ms.)

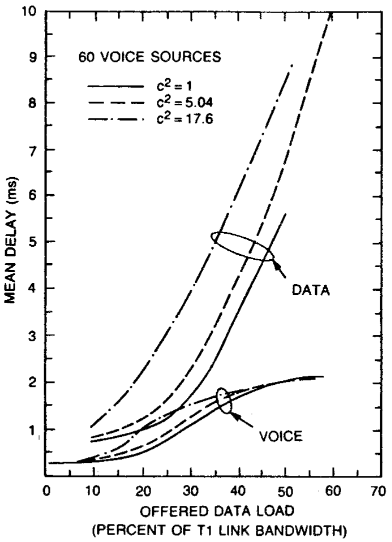

FIGURE 14. Effect of data burstiness on voice and data delays. (Parameter values same as in Fig. 13.)

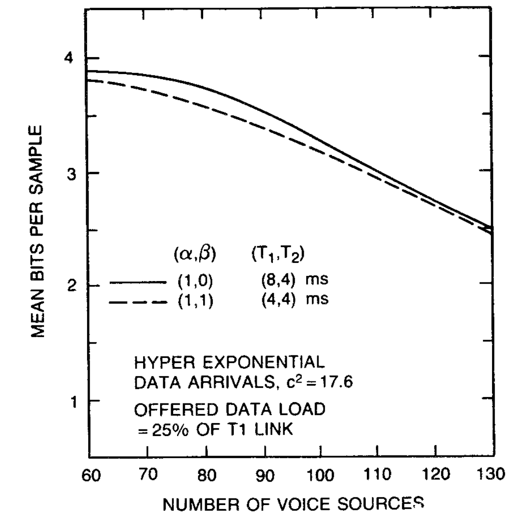

FIGURE 12. Sensitivity of mean bits per sample for β = 0 *vs.* β = 1.

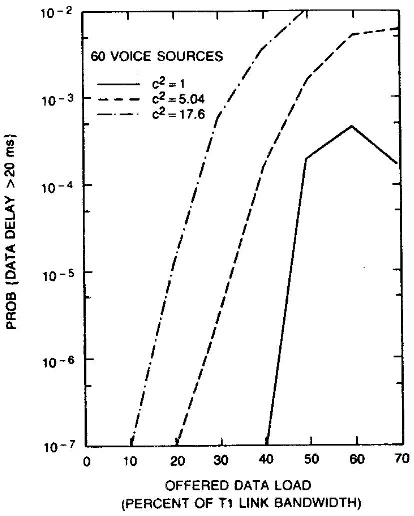

FIGURE 15. Effect of data burstiness on tail probability for data delay. (Parameter values same as in Fig. 13.)